

**Weiterbildender Fernstudiengang Angewandte
Umweltwissenschaften**

Systemverwaltung in der Umweltwissenschaft

Kurseinheit:

Umweltdatenbanken und Umweltinformationssysteme

Autoren:

Prof. Dr. Reiner Güttler
Fachbereich Grundlagen - Informatik - Sensortechnik
Hochschule für Technik und Wirtschaft des Saarlandes

&

Dipl.-Phys. Ing., Dipl.-Umweltwiss. Bernd H. Schlüter
Haltern am See

Inhaltsverzeichnis

1. Einleitung	6
1.1 Informationen zur Kurseinheit	6
1.2 Umweltinformatik - Rolle von Datenbanken.....	8
2. Grundlagenwissen über Datenbanken	11
2.1 Grundprinzipien.....	11
2.2 Relationale Datenbanken.....	19
2.2.1 Grundlagen des Relationenmodells.....	19
2.2.2 Die Datenmanipulationssprache SQL	21
2.2.3 Kopplungsvarianten zu Programmiersprachen.....	23
2.2.4 Integritätssicherung/Datenintegrität	24
2.2.5 Views	26
2.2.6 Datenschutz und Zugriffskontrolle.....	26
2.2.7 Transaktionsverwaltung	28
2.2.8 Informationsmodellierung mit Semantischen Datenmodellen....	31
2.2.9 Werkzeuge zur Anwendungsentwicklung	39
2.2.10 Verteilte Datenbanken - Client-Server-Systeme.....	40
2.2.11 Wichtige Produkte (relationale Datenbanksysteme)	42
2.3 Objektorientierte Datenbanken.....	43
2.3.1 Schwachpunkte relationaler Datenbanksysteme.....	44
2.3.2 Grundkonzepte objektorientierter Datenbanksysteme.....	46
2.4 Gegenüberstellung relationale/objektorientierte Datenbanken.....	48
2.5 Objektrelationale Datenbanken.....	49
2.5.1 Einführung zu ORDBS	49
2.5.2 SQL-Standard SQL-99 und SQL-2003.....	50
2.6 Gegenüberstellung zum Datenbankeinsatz	50
2.7 Neue Datentypen.....	51
2.7.1 IRS/Volltext-Datenbanken.....	51
2.7.2 Datentypen für multimediale Anwendungen	54
2.7.3 Raumbezogene Datentypen, Operatoren und Indizierung	54
3. Daten und Informationen im Umweltbereich	55
3.1 Rolle von Umweltinformationen	55
3.2 Von Daten zu Informationen.....	55
3.2.1 Datenerhebung	56
3.2.2 Datenaufbereitung.....	56
3.2.3 Datenhaltung.....	60
3.2.4 Datenanalyse.....	60
3.3 Besonderheiten von Umweltdaten und -informationen.....	60
4. Aufbau von Umweltinformationssystemen	64
4.1 Allgemeine Anforderungen	64
4.2 Komponenten von Umweltinformationssystemen.....	68

4.3	Klassifizierung von UIS.....	69
4.3.1	Klassifizierung aus inhaltlicher Sicht.....	70
4.3.2	Klassifizierung nach Organisationsstrukturen.....	72
4.3.3	Klassifizierung nach Informatikgesichtspunkten	73
4.4	Architektur von UIS	76
4.5	Besondere Probleme	77
4.5.1	Verteilung - Integration - Heterogenität	77
4.5.2	Autonomie.....	82
4.5.3	Dynamik.....	83
4.6	Metadaten – Datenkataloge.....	85
4.6.1	Typen von Metainformationen	85
4.6.2	Datenkataloge als Metainformationssysteme.....	87
4.6.3	Anforderungen an Umweltdatenkataloge.....	88
4.6	Weitere Entwicklung von UIS	89
4.8	Beispiele für UIS	93
4.8.1	Umweltdatenkatalog UDK des Bundes und der Länder	93
4.8.2	Umweltinformationssysteme der Bundesländer	93
4.8.3	Kommunale Umweltinformationssysteme	97
5.	Betriebliche Umweltinformationssysteme	104
5.1	Grundlagen.....	104
5.2	Informationsbedarf	106
5.2.1	Interner Informationsbedarf.....	106
5.2.2	Externer Informationsbedarf	108
5.3	Aufgabenfelder	108
5.3.1	Umweltrelevante Informationen im Unternehmen.....	110
5.3.2	Modellierung von Stoff- und Energieflusssystemen.....	110
5.3.3	Ökobilanzierung mit BUIS.....	111
5.3.4	Produktionsnahe BUIS.....	113
5.3.5	Umweltberichterstattung	116
5.3.6	Zwischenbetriebliche Logistik und Verwertungsverbunde	117
5.4	Konzeption und Aufbau von BUIS	119
5.4.1	BUIS im Kontext betrieblicher Informationssysteme.....	120
5.4.2	Softwareprodukte für BUIS	120
6.	Allgemein zugängliche Umweltinformationen.....	123
6.1	Übersicht.....	123
6.2	Nutzung im Umweltbereich.....	125
6.3	Suche nach Informationsquellen.....	127
7.	Literaturverzeichnis	131
8.	Übungsaufgaben.....	132
8.1	Aufgabenstellungen	132
8.1.1	Übungsaufgaben zu Kapitel 2	132
8.1.2	Übungsaufgaben zu Kapitel 3	133
8.1.3	Übungsaufgaben zu Kapitel 4	134

8.1.4	Übungsaufgaben zu Kapitel 5.....	135
8.1.5	Übungsaufgaben zu Kapitel 6.....	135
8.2	Lösungen.....	136
8.2.1	Lösungen zu Kapitel 2.....	136
8.2.2	Lösungen zu Kapitel 3.....	138
8.2.3	Lösungen zu Kapitel 4.....	139
8.2.4	Lösungen zu Kapitel 5.....	141
8.2.5	Lösungen zu Kapitel 6.....	142
Index		143

Abkürzungs- und Akronymverzeichnis

2PC	2-Phasen-Commit-Protokoll
4GL	4th Generation Language
ADT	Abstrakter Datentyp
ALBIS	Arten-, Landschafts- und Biotop-Informationssystem
ALK	Automatisierte Liegenschaftskarte
ASIS	Abfall- und Sonderabfallinformationssystem
ATKIS	Amtlich topografisch-kartografisches Informationssystem
BUIS	Betriebliche Umweltinformationssysteme
CAD	Computer-Aided Design
CIB	Computer Integrated Business
CIM	Computer-Integrated Manufacturing
CPU	Central Processing Unit
DB	Datenbank
DBS	Datenbanksystem
DIM	Daten- und Informationssystem
DORIS	Dortmunder Raumbezogenes Informationssystem
DSS	Decision Support Systems
DTD	Dokumenttypdefinition
DV	Datenverarbeitung
EG	Europäische Gemeinschaft
EH&S	Environment, Health and Safty
EMAS	Eco-Management and Audit Scheme
EPA	Environmental Protection Agency
ERIN	Environmental Resources Information Network
ERM	Entity-Relationship-Modell
ESQL	Embedded SQL
EU	Europäische Union
EUR	Euro
EUS	Entscheidungsunterstützende Systeme
FTP	File Transfer Program
gein	German Environmental Information Network
GI	Gesellschaft für Informatik
GIS	Geografisches Informationssystem
html	hypertext mark-up language
http	hypertext transfer protocol

IRS	I nformation- R etrieval- S ystem
KIS	K ommunales I nformationssystem
KNN	K ünstliche n euronale N etze
KUIS	K ommunales U mweltinformationssystem
KVR	K ommunalverband R uhrgebiet
LCA	L ife C ycle A ssessment
LIPS	L abor- I nformations- und P lanungssystem
LIS	L and- I nformationssystem
LfU	L andesanstalt für U mweltschutz. B aden W ürttemberg
MB	M ega B yte
MURL	M inisterium für U mwelt, R aumplanung und L andwirtschaft, N RW
NIC	N etwork I nformation C enter
NUMIS	N iedersächsisches U mweltinformationssystem
OODBS	O bjektorientierte D aten b anksysteme
ORDBS	O bjektrelationale D aten b anksysteme
PLM	P roduct L ifecycle M anagement
PPS	P roduktionsplanung- und -steuerung
PRPS	I ntegrierte P roduktions- und R ecyclingplanung und -steuerung
RIPS	R äumliches I nformations- und P lanungssystem
RIS	R auminformationssystem
SQL	S tructured Q uery L anguage
TCP/IP	T ransmission C ontrol P rotocol/ I nternet P rotocol
TP	T ransaction P rocessing
TULIS	T echnosphären- und L uftinformationssystem
UDK	U mwelt d aten k atalog
UDO	U mweltinformationssystem D ortmund
UFIS	F ührungsinformationssystem
UIS	U mweltinformationssystem
UML	U nified M odeling L anguage
UNO	U nited N ations O rganization
UQUADO	D igitale U mwelt q ualitätskarte D ortmund
UVP	U mweltverträglichkeitsprüfung

1. Einleitung

Lehrziele zu Kapitel 1

Nach Erarbeiten des Kapitels 1 sollten Sie neben dem Grundlagenwissen über die Kurseinheit selbst (Aufbau, Inhalte und Gesamtlehrziele) wissen bzw. erklären können,

- ◆ dass es im Umweltbereich mehr **datenverwaltungsorientierte** und mehr **verarbeitungsorientierte** Informationssysteme gibt,
- ◆ dass im Umweltbereich auch die mehr verarbeitungsorientierten Systeme meist ihre Informationen aus **Datenbanken** beziehen,
- ◆ dass andere Datenverwaltungsmechanismen für 'normale' Daten **i.d.R.** weit höhere Betriebs- und Wartungskosten verursachen,
- ◆ dass für bestimmte Typen von Daten, die im Umweltbereich häufig vorkommen, neue Typen von **Datenverwaltungsmechanismen** notwendig sind,
- ◆ dass im Umweltbereich häufig **große Datenmengen aus unterschiedlichen Systemen** zusammengeführt werden müssen,
- ◆ dass aufgrund der Nutzerstruktur der Zugriff auf die Daten sehr einfach sein muss,
- ◆ dass in zunehmendem Maße die **Offenheit** der Informationssysteme Bedeutung erhält.

1.1 Informationen zur Kurseinheit

Inhalte

Die Kurseinheit 'Umweltdatenbanken und Informationssysteme' soll Ihnen einen Überblick über das weite Feld der **Umweltinformationssysteme** geben. Der Begriff ist in Wissenschaft und Praxis nicht exakt definiert. Der Studienbrief soll Ihnen das notwendige Grundlagenwissen aus dem Umfeld all dessen vermitteln, was allgemein unter Umweltinformationssystem gehandelt wird und Ihnen somit eine Einordnung konkreter Systeme erlauben. Zu diesem Grundlagenwissen gehört aus informationstechnischer Sicht auch ein Mindestmaß an Wissen über die **Datenbanktechnologie**.

Voraussetzungen

Für die Kurseinheit sollten Sie grob mit den wichtigsten Begriffen aus den Umweltwissenschaften vertraut sein. Organisatorische, politische und juristische Strukturen im Umweltbereich sollten Ihnen einigermaßen vertraut sein.

Aus informationstechnischer Sicht genügt völlig ein reines 'Anwenderwissen', d.h. sie sollten mit den wichtigsten Begriffen dessen, wie sich ein Computersystem samt Anwendungsprogramm einem Benutzer präsentiert, vertraut sein.

Zu den einzelnen Kapiteln der Kurseinheit gibt es **Übungsaufgaben samt Musterlösungen**. Diese Aufgaben bestehen im Wesentlichen aus Verständnisfragen. Konstruktive Aufgaben im Sinne des eigentlichen Aufgabenbegriffs wären allenfalls bei enormer Ausweitung des Informatikaspekts in dieser Kurseinheit möglich. Dies aber würde den allgemeinen Zielen des Kurses widersprechen. Zu den einzelnen Kapiteln sind jeweils Lehrziele aufgelistet, deren Erreichen Sie u.a. durch die Übungsaufgaben überprüfen können.

Die globalen Lernziele der gesamten Kurseinheit können wie folgt zusammengefasst werden:

Globale Lernziele

- ◆ Verständnis der Rolle der **Datenbanktechnologie** und (häufig darauf aufbauender) Informationssysteme im Umweltschutz (Kap.1),
- ◆ Vermittlung des notwendigen Grundlagenwissens über **Datenbanken und Informationssysteme** aus informationstechnischer Sicht (Kap.2),
- ◆ Kenntnis der Bedeutung des Unterschieds zwischen **Daten und Informationen** (Kap.3),
- ◆ Verständnis der **Besonderheiten von Umweltinformationen** im Vergleich zu Informationen in sonstigen Informationssystemen,
- ◆ und daraus resultierender allgemeiner Anforderungen an **Umweltinformationssysteme** (Kap.4.1),
- ◆ Kenntnis der **Klassifizierungen** von Umweltinformationssystemen nach verschiedenen Kriterien (thematisch, aus behördlicher Sicht, aus informationstechnischer Sicht usw.) sowie konkret vorliegende Beispiele zu den verschiedenen Klassifizierungen (Kap. 4.3),
- ◆ Verständnis der **Architektur** von Umweltinformationssystemen und Auswirkungen auf die dabei notwendigen Datenbanken (mit Schwerpunkt Dezentralisierung, verteilte Systeme, verteilte Datenhaltung/Datenbanken) (Kap. 4.4),
- ◆ Erläuterung besonderer inhaltlicher **Probleme und Lösungsansätze** (z.B. **Metainformationssysteme** bei der Integration von vorhandenen Insellösungen zu einem Umweltinformationssystem, **Umweltdatenkataloge**, Einhaltung von Standards u.ä.) (Kap. 4.5),
- ◆ Darbietung einer Übersicht mit **Beispielen** zu Umweltinformationssystemen des Bundes, der Länder und der Kommunen (Kap. 4.8),
- ◆ Kenntnisse über Einsatz, Besonderheiten und Weiterentwicklung von **Betrieblichen Umweltinformationssystemen** mit Beispielen (Kap. 5),
- ◆ Vorstellung allgemein zugänglicher **Umweltinformationen**; Informationen im WWW,
- ◆ Verständnis der Abgrenzung von Umweltdatenbanken und datenbankgestützten Umweltinformationssystemen (Kap. 6).

Anmerkungen zur 3. und 4. Auflage:

Die vorliegende Kurseinheit wurde vollständig überarbeitet und fachlich auf den aktuellen Stand gebracht. Insbesondere haben sich die Konzeption (vergleiche Kap. 4.7) sowie die Einsatz- und Nutzungsmöglichkeiten von Umweltinformationssystemen (Beispiele zu UIS, Kap. 4.8) in den letzten Jahren weiterentwickelt. Dieses trifft auch für die betrieblichen Umweltinformationssysteme zu, die jetzt in Kapitel 5 ausführlicher behandelt werden. Dieses ist u.a. auf neue DV-Technologien (z.B. im Internet) zurückzuführen und bietet neue Zugangsmöglichkeiten zu Umweltinformationen (siehe Kap. 6).

Zur Verbesserung der 'Lesbarkeit' der Kurseinheit wurden die Kapitel zum Teil neu strukturiert, wichtige Definitionen und Merksätze in einen **Rahmen** gesetzt sowie bei langen Textpassagen **informative Wörter und Satzteile fett markiert**. Zudem sind viele Links ins Internet eingerichtet, über die auf unkomplizierte Weise weitergehende und vertiefende Informationen (z.B. <http://www.uni-koblenz.de/~wfu/>) abgerufen werden können. Am Anfang der Kurseinheit ist jetzt ein Abkürzungs- und Akronymverzeichnis vorhanden und am Ende ein Index.

In der 4. Auflage wurde im Kapitel 2 'Grundlagenwissen über Datenbanken' der aktuelle Entwicklungsstand der heute verfügbaren objektrelationalen Datenbanksysteme aufgezeigt.

1.2 Umweltinformatik - Rolle von Datenbanken

Probleme können ohne die Information über sie nicht gelöst werden. Komplexe Probleme benötigen zu ihrer Lösung i.d.R. mehr Informationen als einfache Probleme. Umweltprobleme sind fast immer komplex.

Informatikwerkzeuge im Umweltschutz

Diese einfachen Tatsachen verlangen zwingend den Einsatz **rechnergestützter Informationsverarbeitung** bei der Bewältigung der mannigfaltigen Aufgaben, die uns durch die sich verschärfende Umweltproblematik überall gestellt werden. Folgende Aspekte sind dabei besonders relevant für eine computergestützte Informationsverarbeitung, dazu angegeben jeweils die entsprechenden **Informatikwerkzeuge**:

- ◆ Verwaltung großer Datenmengen: **Datenbanken**,
- ◆ Datenauswertung: statistische und andere **Auswertungs- und Darstellungsprogramme** für große Datenmengen,
- ◆ räumliche Differenzierung: **geografische Informationssysteme**,
- ◆ zeitliche und räumliche Entwicklung: z.B. **Simulationssysteme**,
- ◆ Szenarienanalyse, Bewertungen, Folgenabschätzungen, Entscheidungsunterstützung: **wissensverarbeitende Systeme, Expertensysteme**,
- ◆ Erfahrung von Zusammenhängen und Dynamiken: **Visualisierung**,

- ◆ risikoloses Experimentieren (Parameter- und Strukturänderung): **Computer-simulation**.

Alle diese Bereiche spielen eine wichtige Rolle, wenn auch zur Zeit noch der Aspekt des Sammelns, Speicherns und Manipulierens von Daten überwiegt. Wichtig ist festzuhalten, dass aber auch die Systeme, die weniger datenverwaltungs- sondern mehr **verarbeitungsorientiert** sind, für ihr korrektes Funktionieren auf einem gesicherten Informationsstand aufbauen müssen. Hierzu beziehen sie i.a. ihre Eingabedaten aus den Datenverwaltungssystemen. Aus gutem Grund liegt heute den meisten dieser Systeme ein **Datenbanksystem** (DBS) zu Grunde. Wesentliche Gründe hierfür sind u.a.:

Datenbanksysteme

- ◆ die Möglichkeit, mit relativ **kurzen Entwicklungszeiten** sehr einfach zu bedienende Anwendungsprogramme zu entwickeln
- ◆ die enorm **verringerten Wartungskosten** für die Datenverwaltung selbst (im Vergleich zu anderen Datenverwaltungssystemen)

Kapitel 2 wird diesen Sachverhalt näher beleuchten.

Als **Umweltdatenbanken** werden i.a. jene Datenbanken bezeichnet, die gezielt zur (interaktiven) Beantwortung umweltbezogener Fragen und/oder um umweltbezogene Anwendungen (z.B. Auswerte- und Simulationsprogramme im Umweltbereich) mit Daten zu versorgen aufgebaut sind. Datenbanken, die nur zufällig auch Umweltdaten enthalten, fallen nicht in die Rubrik Umweltdatenbanken.

Umweltdatenbanken

Umweltinformationssysteme bauen i.d.R. auf Umweltdatenbanken auf. Eine klare Abgrenzung der Begriffe Umweltinformationssystem und Umweltdatenbank ist nicht möglich. Der Übergang von einer Umweltdatenbank zu einem Umweltinformationssystem ist fließend, auch werden häufig Umweltdatenbanken als Umweltinformationssysteme (und umgekehrt) bezeichnet.

Umweltdaten sind **raum- und zeitbezogene Daten** zu verschiedenen **Umweltmedien** (Luft, Wasser, Boden) sowie einer Vielzahl von **Problembereichen** (z.B. Abfall, Lärm, gefährliche Stoffe, Landschafts- und Naturschutz usw.). In allen genannten Bereichen spielen Gesichtspunkte wie Messbarkeit, Menge, Intensität und die Wirksamkeit auf Mensch, Tier, Pflanze u.a. eine zentrale Rolle. Durch Analysen und Interpretationen solcher **Daten** werden **Informationen** über unsere Umwelt gewonnen.

Daten und
Informationen

Kapitel 3 beschäftigt sich mit dem Zusammenhang zwischen **Umweltdaten und -informationen**.

Umweltinformati-
onssysteme

Je mehr Verständnis über Zusammenhänge in unserer Umwelt gewonnen werden, um so mehr entsteht ein Bedarf an der **Zusammenführung überregionaler und interdisziplinärer Daten**. Das Auffinden von relevanten Daten sowie die **Kommunikation** gefundener Daten erlangen immer größere Bedeutung. Datenquellen bzw. teilweise auch **Informationsquellen** sind i.a. Ämter, Regional- und Landesregierungen, Forschungszentren, Planungsbüros usw.. **Umweltinformationssysteme** und damit zusammenhängende **Umweltdatenkataloge** sollen den Zugang zu den Informationen in den verschiedenen räumlich und inhaltlich getrennten Systemen erleichtern. Dem Aspekt der **Offenheit der Systeme** kommt immer größere Bedeutung zu. Hierbei sind viele Fortschritte erzielt (z.B. im Bereich Vernetzung), es bleiben derzeit jedoch noch viele **Probleme** offen. Die Thematik der Umweltinformationssysteme wird in Kapitel 4 behandelt.

Betriebliche
Umweltinforma-
tionssysteme

Neben den Umweltinformationssystemen der öffentlichen Verwaltung hat sich in den letzten Jahren der Einsatz von **Betrieblichen Umweltinformationssystemen (BIUS)** zur Verarbeitung umweltrelevanter Informationen im Betrieb weiter entwickelt. Kapitel 5 stellt auf Basis des **betriebsinternen/-externen Informationsbedarfs** die Aufgabengebiete von BIUS vor. Hier wird die Anwendung von BIUS u.a. für die **Modellierung von Stoff- und Energieflüssen**, für die **Ökobilanzierung** und im **produktionsnahen Einsatz** dargestellt.

Internet

Kapitel 6 beschäftigt sich mit allgemein oder auch **öffentlich zugänglichen Umweltdatenbanken und -informationssystemen**. Datenbankanbieter bieten über Telekommunikationseinrichtungen wie z.B. dem **Internet** Informationen mit großer inhaltlicher Breite (z.B. Stoff- und Faktendatenbanken, Literaturdatenbanken, juristische Datenbanken usw.) an. Es gibt dabei kommerziell und nicht kommerziell betriebene Systeme, dazu sogenannte Informationsvermittler zur Orientierung.

Umweltdatenbanken, die übers Internet erreicht werden können, werden in Kapitel 6 vorgestellt. Zudem wird der Einsatz von **Suchsystemen** zur Auffindung von Informationsquellen im Internet erläutert.

2. Grundlagenwissen über Datenbanken

Lehrziele zu Kapitel 2

Nach Erarbeiten von Kapitel 2 sollten Sie in der Lage sein:

- ◆ wesentliche Begriffe und **Grundprinzipien von Datenbanken** allgemein aufzählen und zu erläutern,
- ◆ Komponenten, Aufbau und Funktionsprinzip von **relationalen Datenbanken** grob zu erläutern,
- ◆ zu erklären, was der **objektorientierte Ansatz** (Denkweise) ist,
- ◆ **Vorteile, Anwendungsgebiete und spezielle Datentypen von objektrelationalen Datenbanken** zu kennen,
- ◆ sie sollten wissen, dass es neben Datenbanken auch andere Basiskomponenten für die **Speicherung von Informationen** gibt, sie sollten deren Grundprinzip (kurz) erläutern können,
- ◆ sie sollten den **Aufbau von Informationssystemen** und die Rolle von Datenbanken und anderen Basiskomponenten darin erklären können,
- ◆ sie sollten wesentliche Prinzipien der Entwicklung von **Datenbankanwendungsprogrammen** erläutern können sowie die Unterstützung durch Entwicklungswerkzeuge,
- ◆ sie sollten wissen und grob erklären können, was ein **Entity-Relationship-Modell** ist,
- ◆ sie sollten mit den Grundprinzipien **verteilter Datenbanken** und **Client-Server-Systemen** grob vertraut sein,
- ◆ sie sollten einige weit verbreitete Datenbank-Produkte aufzählen können.

2.1 Grundprinzipien

Datenbanksysteme bilden das Softwarerückgrat der meisten computergestützten Informationssysteme. Sie stellen dem Anwendungsentwickler generische **Operationen** zum Suchen in sehr großen Datenbeständen sowie zum Einfügen, Ändern und Löschen von Daten zur Verfügung. Operationen werden in einer deskriptiven **Datenmanipulationssprache** spezifiziert, meist in der **standardisierten Sprache SQL** (Structured Query Language), und die **Operationsaufrufe** werden in Anwendungsprogramme eingebettet, die in einer konventionellen Programmiersprache oder einer sogenannten '4th Generation Language' (4GL) geschrieben sind. Die folgenden Beispiele nennen einige typische Informationssysteme, die auf Datenbanksystemen beruhen.

- ◆ Verwaltung eines Teilelagers eines Automobilherstellers. Dies umfasst u.U. mehrere Millionen Teile für alle Modelle, für die der Hersteller die Lieferung von Ersatzteilen garantiert. Das System dient u.a. zur Buchführung

über die Lagerbestände und zum Auffinden der Teile in roboterbedienten Lagern. Zudem gibt das System Auskunft darüber, welche Teile zu welchen Modellen gehören, welche Teile untereinander kompatibel sind, usw.

- ◆ Entscheidungsunterstützung für den Wertschriftenhandel einer Bank. Börsenkurse und damit zusammenhängende Informationen (z.B. Wirtschaftsnachrichten) werden über entsprechende Datenleitungen von verschiedenen Börsen kontinuierlich erfasst, und es werden komplexe Suchoperationen auf den Daten spezifiziert, die die Händler auf interessante Konstellationen aufmerksam machen sollen. Dies ist ein Beispiel eines Echtzeit-Informationssystems mit harten Anforderungen an die Antwortzeit der Suchoperationen und extrem hohen Änderungsraten der Daten (z.T. mehr als 100 Kurse pro Sekunde).
- ◆ Verwaltung von Patientendaten in einem Krankenhaus. Dies beinhaltet Informationen über die Krankengeschichte eines Patienten, Untersuchungsergebnisse, Einsatz von Medikamenten, Behandlungen, bis hin zur Kostenabrechnung. über die für administrative Zwecke notwendigen Such- und Änderungsoperationen hinaus ist u.a. die Unterstützung der Suche nach ähnlichen Krankheitsbildern wünschenswert. Zusätzlich wird die Anbindung von Archivsystemen für Ultraschall- und Röntgenbilder, Computertomografien u.ä. angestrebt.

Grundprinzipien
von Datenbanksystemen

Aus den aufgeführten Anwendungen lassen sich verschiedene Anforderungen an Datenbanksysteme ableiten, die zu **vier Grundprinzipien** führen:

- ◆ Optimierung von Datenspeicherung und -zugriff,
- ◆ Unabhängigkeit von Daten und Anwenderprogrammen,
- ◆ Sicherstellung der Datenkonsistenz,
- ◆ Datenbankänderungen nach dem Transaktionsprinzip.

Datenbanksysteme haben u.U. **riesige Datenmengen** zu verwalten. Typische Datenbankgrößen in kommerziellen oder administrativen (Umweltbereich!) Anwendungen liegen im Bereich zwischen 10 und einigen 100 Gigabytes, sehr große Datenbanken umfassen mehr als 1 Terabyte. Daraus folgt zwangsläufig, dass Datenbanken in erster Linie auf **Sekundärspeichermedien** liegen, in aller Regel auf Magnetplatten (Festplatten/-systemen). Die Datenmengen können nicht oder nur in geringem Umfang im Hauptspeicher des Computersystems gehalten werden.

Speicherung

Zwischen Hauptspeicher und Magnetplatten aber klafft ein großer **Geschwindigkeitsunterschied** von einigen Zehnerpotenzen. In Verbindung mit der Nebenbedingung, dass Daten auf Magnetplatten nur blockweise adressiert werden können, ergibt sich die Notwendigkeit, dass Datenbanken anders als mit klassischen, hauptspeicherorientierten Datenstrukturen organisiert werden. Diese Überlegung führt auf das folgende Grundprinzip.

Grundprinzip 1:

Um große Datenmengen effizient verwalten zu können, sind Datenbanksysteme im Hinblick auf **Sekundärspeicherzugriffe** optimiert.

Insbesondere sind auch die klassischen Modelle der Komplexitätstheorie für Datenbanksysteme nur von untergeordneter Bedeutung; die entscheidende Effizienzmetrik ist vielmehr die **Anzahl der Sekundärspeicherzugriffe**.

Da Datenbanksysteme **generische Systeme** sind, also gleichermaßen in Bankkonten wie auch in Patienten- oder Altlastendaten suchen und ändern können, sind universelle Optimierungen der Systemeffizienz nur eingeschränkt möglich. Vielmehr hängt der **Nutzen einer Optimierungsmaßnahme häufig von den spezifischen Lastmerkmalen der jeweiligen Anwendung ab**. In einer Anwendung mit geringer Änderungsrate kann man Daten - ggf. unter Einführung von Redundanz - so speichern, dass verschiedenartige Suchoperationen extrem effizient ausgeführt werden können; in einer Anwendung mit hoher Änderungsrate hingegen kann dies unakzeptabel sein, wenn die Aktualisierung der entsprechenden Datenstrukturen zu aufwändig ist.

Aus diesem Grund unterstützt ein Datenbanksystem in der Regel ein breites Spektrum **verschiedener Speicherungs- und Zugriffsstrukturen** und überlässt es einem Tuningspezialisten, diese Strukturen für eine Anwendung festzulegen. Da sich aber Lastmerkmale einer Anwendung im Laufe der Zeit ändern (z.B. durch Erweiterung der Anwendungsfunktionalität), müssen diese Tuning-Entscheidungen hin und wieder revidiert werden.

Tuning

Solche Änderungen der Speicherungs- und Zugriffsstrukturen sind nicht gerade eine billige Maßnahme, da dazu häufig große Datenmengen vom Datenbanksystem umgespeichert werden müssen. Von fundamentaler Bedeutung ist jedoch, dass bei einer solchen Revision der Datenspeicherung die Anwendungsprogramme auf keinen Fall geändert werden müssen, denn der entsprechende Änderungsaufwand wäre kostenmäßig in den meisten Fällen absolut unakzeptabel, insbesondere im Umweltbereich aufgrund der hohen Dynamik (siehe auch Kap. 4.5.3.) und weil fast alle Entwicklungsvorhaben extern vergeben werden.

Grundprinzip 2:

Die Speicherung von Daten kann geändert werden, ohne dass Anwendungsprogramme geändert werden müssen (**Programm-Daten-Unabhängigkeit**).

Programm - Daten-
Unabhängigkeit

Datenbanksysteme stellen diese sogenannte **Programm-Daten-Unabhängigkeit** (häufig auch nur: **Datenunabhängigkeit**) sicher, indem sie die eigentlichen Speicherungs- und Zugriffsstrukturen vor dem Anwendungsprogramm verbergen. Mit anderen Worten: Datenbanksysteme wenden das **Prinzip der Abstrakten Datentypen** auf generische Weise an, indem sie die verwalteten Daten kapseln und nur deskriptive Operationen anbieten, die unabhängig von den zu Grunde liegenden

konkreten Datenstrukturen sind.

In eingeschränkter Form gilt dieses Prinzip sogar auch bei einer anwendungsbedingten **Änderung des Datenformats** (z.B. der Umstellung der Postleitzahlen in Deutschland) sowie bei der nachträglichen **Erweiterung existierender Daten um zusätzliche Informationen** (z.B. der Hinzunahme einer Kreditkartennummer für alle Privatpatienten einer Klinik, zusätzliche Parameter in einer Untersuchungsliste usw.).

Dieses fundamentale Prinzip hat ganz entscheidend zum **kommerziellen Erfolg** von Datenbanksystemen beigetragen. Etliche Informationssysteme, die ohne echtes Datenbanksystem entwickelt wurden (also beispielsweise auf der Basis eines File-Systems), sind mittelfristig gescheitert, weil sie aufgrund fehlender Programm-Daten-Unabhängigkeit unakzeptable Programmwartungskosten verursachten.

Das Grundprinzip der Programm-Daten-Unabhängigkeit und die damit einhergehende Konzeption **deskriptiver Datenmanipulationssprachen** wurde vor allem durch die Entwicklung **relationaler Datenbanksysteme** in den Siebziger und Achtziger Jahren entscheidend vorangebracht. Bei diesen Systemen, die heute den Markt dominieren, werden alle Daten konzeptionell in Form von **Tabellen** (mathematisch: **Relationen**), repräsentiert.

Einen Schritt weiter geht man seit einigen Jahren mit **objektorientierten Datenbanksystemen** (bzw. entsprechenden 'postrelationalen' Weiterentwicklungen relationaler Systeme). Diese Systemklasse erlaubt es insbesondere, **Aggregationsbeziehungen zwischen Datenelementen** in Form '**komplexer Objekte**' direkt zu repräsentieren. Beispielsweise können alle Teile eines Motors zu einem Objekt 'Motor' zusammengefasst werden, alle Teile einer Industrieanlage zum Objekt 'Anlage', alle Parameter einer Wasserprobe einschließlich der Probenstellenbeschreibung zu einem Objekt 'Wasserprobe' - eine Form der Aggregation, die mit relationalen Systemen nur auf umständliche Art möglich ist.

Objektorientierte
Datenbanken

Generische Such- und Änderungsoperationen werden sowohl auf der Ebene der Gesamtobjekte als auch auf der Ebene der Teilobjekte angeboten. Darüber hinaus ist es in objektorientierten Datenbanksystemen auch möglich, **anwendungsspezifische Operationen** für eine Menge von Objekten zu definieren. Diese können dann in Ausdrücken der Datenmanipulationssprache verwendet werden.

Das Datenbanksystem, welches an sich als generisches Softwarepaket konzipiert ist, wird damit **anwendungsspezifisch erweiterbar**. Beispielsweise kann man eine Funktion 'Volumen' für geometrische Objekte definieren, die auf Polygonen für die Begrenzungsflächen eines dreidimensionalen Körpers operiert. Wie in objektorientierten Programmiersprachen können diese Funktionen mittels '**Vererbung**' auf flexible Art und Weise für speziellere Objekte wiederverwendet werden (z.B. für die speziellen Körper, die beim mechanischen CAD auftreten); da-

bei können bei Bedarf spezifische Details 'überschrieben', d.h. neu festgelegt werden (z.B. die Interpolationsvorschrift zwischen Stützpunkten eines Polygons). Das Prinzip der Programm-Daten-Unabhängigkeit sollte auch für objektorientierte Datenbanksysteme unter Einschluss der anwendungsspezifischen Operationen gelten. Allerdings gibt es durchaus noch offene Fragen in diesem Zusammenhang, die Gegenstand aktueller Forschung sind, beispielsweise die Konzeption und effiziente Implementierung objektorientierter Datenmanipulationssprachen und deren Einbettung in objektorientierte Programmiersprachen.

Grundprinzip 3:

Die **Konsistenz der Daten** wird vom Datenbanksystem gewährleistet.

Die Effektivität eines Informationssystems steht und fällt mit der **Korrektheit der zu Grunde liegenden Daten**. Eine Form der Datenkorrektheit, die von Datenbanksystemen sehr weitreichend unterstützt wird, ist die **Konsistenz** der Daten (oder auch: **Integrität** der Daten). Diese fordert, dass bestimmte Zusammenhänge zwischen verschiedenen Datenelementen, die in der realen Welt gelten sollen, auch in der Datenbank gewährleistet werden. Beispielsweise soll das Konto eines Kunden bei einem Versandhaus nur dann belastet werden, wenn eine entsprechende Bestellung ausgeliefert wird, ein bestimmter Alarm nur ausgelöst werden, wenn ein bestimmter Messwert einen bestimmten Grenzwert überschritten hat.

Datenbanksysteme bieten die Möglichkeit an, dass solche **anwendungsspezifischen Konsistenzbedingungen** spezifiziert werden, und die Systeme überwachen die Einhaltung dieser Bedingungen. Änderungen, die die Konsistenz verletzen würden, werden zurückgewiesen. Damit werden gewissermaßen die **Systeme 'immun' gegen fehlerhafte bzw. inkonsistente Dateneingaben**.

Diese Vorgehensweise ist zwar grundsätzlich auf **Plausibilitätsprüfungen** beschränkt, da theoretisch auch fehlerhafte Eingaben formal konsistent sein können; in vielen Anwendungen aber können dadurch fast alle Eingabefehler erkannt und eliminiert werden.

Einige Klassen von Konsistenzbedingungen können leichter spezifiziert und vom System überwacht werden, wenn die Datenbank selbst daraufhin entworfen wurde. In diese Klasse fallen beispielsweise Bedingungen vom Typ 'keine zwei Konten dürfen dieselbe Kontonummer haben' oder 'die bei einer Buchung angegebene Kontonummer muss in der Datenbank existieren'.

Konsistenz,
Integrität

Die Datenbankforschung hat verschiedene Methoden des sogenannten konzeptionellen (oder auch: logischen) **Datenbankentwurfs** hervorgebracht, die in diesem Sinne auf eine hohe Qualitätssicherung für die Daten abzielen. Diese Methoden, z.B. die **relationale Entwurfstheorie** und vor allem der **Entwurf mit 'Entity-Relationship-Modellen'** (ERM) haben in der Praxis weite Verbreitung gefunden.

Datenbankentwurf,
ERM

Moderne Datenbanksysteme gehen noch einen Schritt weiter und erlauben es, anwendungsspezifische Konsistenzbedingungen mit den Sprachmitteln der Datenmanipulationssprache deskriptiv zu spezifizieren. Mit Hilfe sogenannter **aktiver Datenbankmechanismen** (z.B. 'Trigger' und weitergehende Konzepte) ist es ferner möglich, **spezifische Reaktionen beim Erkennen einer potenziellen Konsistenzverletzung** festzulegen; beispielsweise können anstatt einer Zurückweisung der Änderung bestimmte Folgeänderungen automatisch ausgelöst werden. Diese Delegation der Verantwortung für die Datenkonsistenz an das Datenbanksystem stellt eine fundamentale Vereinfachung für die Anwendungsentwicklung dar.

Die Konsistenz der Daten ist nicht nur durch Eingabefehler gefährdet, sondern auch durch **'Interferenzeffekte' paralleler Änderungsoperationen im Mehrbenutzerbetrieb**, insbesondere in verteilter Umgebung (Rechnernetz). Beispielsweise ist es möglich, dass ein Bankkonto 'Geld verliert', wenn etwa zwei Operationen zur Einzahlung von jeweils 100 EUR zunächst beide den Kontostand lesen, jeweils 100 EUR dazu addieren und dann beide denselben Wert in die Datenbank zurückschreiben; die zuerst schreibende Einzahlung wird von der späteren fälschlicherweise überschrieben und geht praktisch verloren. Vermeiden ließe sich dieser Fehler durch entsprechende softwaremäßige Synchronisationsmaßnahmen von Seiten der Anwendungsprogramme; dieses würde aber die Anwendungsentwicklung für Mehrbenutzersysteme erschweren und die Entwicklungsproduktivität verringern.

Ähnliche Fehlereffekte wie der eben beschriebene ergeben sich potenziell auch beim **unkontrollierten Abbruch eines Anwendungsprogramms** oder einem **Ausfall des Datenbanksystems**. Beispielsweise würde eine Überweisung, die nach dem Belasten des ersten Kontos, aber vor der Gutschrift auf das zweite Konto durch einen Fehler abgebrochen wird, die Datenbank inkonsistent hinterlassen und wie zuvor 'Geld verlieren'.

Transaktionskonzept

Datenbanksysteme stellen alle Mechanismen bereit, um Fehlereffekte dieser Art garantiert zu verhindern. Dazu wird vom Anwendungsentwickler verlangt, dass Folgen von Datenbankoperationen in einem Anwendungsprogramm zu **konsistenzhaltenden Einheiten**, sogenannten **Transaktionen**, zusammengefasst und explizit spezifiziert werden. Beispielsweise bilden bei einer Überweisung die Änderungsoperationen beider Konten zusammen eine Transaktion.

Das Datenbanksystem gewährleistet dann die **'(virtuelle) Isolation' von Transaktionen im Mehrbenutzerbetrieb**, indem es (z.B. für das erste Beispiel) entsprechende **Synchronisationsmaßnahmen** für parallele Transaktionen automatisch generiert, und die **'Atomarität'** von Transaktionen, indem die Änderungen unvollständig ausgeführter Transaktionen (wie im zweiten Beispiel) mit Hilfe eines entsprechenden Logbuchs automatisch rückgängig gemacht werden. Mit Hilfe des Logbuchs wird darüber hinaus sichergestellt, dass Änderungen abge-

schlossener Transaktionen wirklich '**dauerhaft**' sind, also weder durch Ausfälle des Datenbanksystems noch durch sonstige Fehler (z.B. Festplattenfehler) verloren gehen. Diese systemgarantierten Eigenschaften von Transaktionen werden auch als **ACID-Prinzip** bezeichnet.

Grundprinzip 4:

Datenbankänderungen werden innerhalb von **Transaktionen** durchgeführt, die atomar (**atomic**), konsistenzertaltend (**consistency-preserving**), isoliert (**isolated**) und dauerhaft (**durable**) sind.

Die Unterstützung von **Transaktionen ist ein Grundpfeiler aller modernen Datenbanksysteme und ein Schlüsselkonzept zur Sicherstellung der Datenkonsistenz**. Die dazu notwendigen Synchronisations- und Fehlertoleranzmaßnahmen werden vom Datenbanksystem sowohl für zentralisierte als auch für verteilte Datenbanken realisiert.

Wegen seiner Allgemeinheit und Leistungsfähigkeit finden das Transaktionskonzept und die entsprechenden Implementierungstechniken der Transaktionsverwaltung auch in Betriebssystemen sowie in Programmiersprachen mit eingebundenen Datenbankfunktionen zunehmende Beachtung; hier werden entsprechende Standards erarbeitet.

Der **Stand der Technik** auf dem Datenbankgebiet lässt sich folgendermaßen zusammenfassen: Datenbanksysteme sind ausgereifte Systeme, ohne die zahlreiche computergestützte Informationssysteme undenkbar wären. Der **Markt** wird heute von relationalen Systemen dominiert; der zurzeit noch kleine Anteil objektorientierter Systeme wächst langsam aber ständig, und viele Anbieter relationaler Systeme sind im Begriff, objektorientierte Konzepte in ihre Systeme zu integrieren.

Weitere Entwicklung

Ungeachtet dieses weit fortgeschrittenen Reifegrads relationaler und - in geringem Maße - auch postrelationaler Datenbanksysteme ist das Datenbankgebiet auch weiterhin ein aktives und wichtiges Forschungsfeld. Die Schwerpunkte der aktuellen Forschung liegen vor allem in einer besseren **Unterstützung komplexer Anwendungen**.

Forschungsschwerpunkte

Technische Herausforderungen sind in diesem Zusammenhang u.a. die Ausnutzung **paralleler Rechenprozesse** zur Beschleunigung komplexer Suchoperationen, bessere Formen der **Einbettung von Datenmanipulationssprachen in Programmiersprachen** mit dem Ziel der Integration beider Welten zu einer 'persistenten (Datenbank-)Programmiersprache' sowie die Konfrontation mit **verteilten und heterogenen Informationssystemlandschaften** mit dem Ziel, selbst unabhängig voneinander entworfene und betriebene Datenbanken oder Informationssysteme übergreifend bearbeiten zu können. Die letztere Forschungsrichtung, stellt zurzeit wohl die größten Herausforderungen für die Datenbankforschung

dar (und ist im Umweltbereich von größter Bedeutung).